**Rezolvarea laboratoarelor**

**Grupa 232 – 1**

**SGBD , 2023 – 2024**

**Forma răspunsurilor**

Nume student – Nr exercițiu

Rezolvare

**Laborator 2**

--4

select category, count(distinct title), count(\*), count(r.copy\_id)

from rental r, title t

where r.title\_id = t.title\_id

group by category

having count(\*) = (select max(count(\*))

from rental r, title t

where r.title\_id = t.title\_id

group by category);

--- Tîrîlă Patric-Gabriel – problema 4

select t2.cat, t2.cnt

from (

select t.category cat, count(\*) cnt

from title t

where t.title\_id in (

select r.title\_id

from rental r

)

group by t.category

) t2

where t2.cnt = (select max(count(\*))

from title t

where t.title\_id in (

select r.title\_id

from rental r

)

group by t.category

)

--Gavrila Alexandru problema 4

select t.category, count(distinct title),count(\*), count(r.copy\_id)

from title t, rental r

where t.title\_id = r.title\_id

group by t.category

having count(\*) =

(select max(count(\*))

from rental r, title t

where r.title\_id = t.title\_id

group by category);

--Codarcea Alexandru problema 4

select t.category, count(distinct t.title\_id)

FROM title t

JOIN rental r ON t.title\_id = r.title\_id

group by category

having count(\*) = (select max(count(\*))

FROM title t

JOIN rental r ON t.title\_id = r.title\_id

group by category);

Ilie Florin-Alexandu -problema 5

select t.title, count(tc.copy\_id) as ex\_disp

from title t

join title\_copy tc on t.title\_id = tc.title\_id

where tc.status like 'AVAILABLE'

group by t.title;

Liviu Firca problema 5

select title\_id,count(title\_id)

from title\_copy

where lower(status)='available'

group by title\_id;

Tîrîlă Patric-Gabriel – problema 5

select t.title, tc2.cnt

from title t, (

select tc.title\_id t\_id, count(\*) cnt

from (select tc3.title\_id title\_id, tc3.copy\_id copy\_id, tc3.status status

from title\_copy tc3

where tc3.status = 'AVAILABLE'

) tc

group by tc.title\_id

) tc2

where t.title\_id = t\_id;

--5 Caraulasu Catalin

select t.title,count(tc.title\_id)

from title\_copy tc, title t

where upper(status)='AVAILABLE' and tc.title\_id=t.title\_id

group by title;

--Codarcea Alexandru problema 5

select t1.title\_id, count(t1.title\_id)

FROM title t1

JOIN title\_copy t2 ON t1.title\_id = t2.title\_id

WHERE upper(status) = 'AVAILABLE'

group by t1.title\_id;

--Gavrila Alexandru problema 5

select title, count(copy\_id)

from title t, title\_copy tc

where t.title\_id = tc.title\_id

and upper(tc.status) = 'AVAILABLE'

group by title;

--Dumitrescul Eduard, problema 5

select count(\*), t.title

from title\_copy tc, title t

where tc.title\_id = t.title\_id

and upper(tc.status) = 'AVAILABLE'

group by t.title;

--5 DOGARU MIHAIL DANUT

select t.title ,count(tc.copy\_id)

from title t,title\_copy tc

where t.title\_id = tc.title\_id

and upper(tc.status) = 'AVAILABLE'

group by t.title;

--5 Papusoi Rares

select count(\*), t.title

from title t, title\_copy c

where t.title\_id = c.title\_id

and lower(c.status) like 'available'

group by t.title;

Firca Liviu problema 6

select t.title,c.copy\_id,c.status,decode(r.act\_ret\_date,null,'Not available','available') Status\_corect

from title t

join title\_copy c on c.title\_id=t.title\_id

join rental r on (r.copy\_id=c.copy\_id and r.title\_id=c.title\_id)

where book\_date=(select max(book\_date)

from rental

where copy\_id=r.copy\_id

and t.title\_id=title\_id);

Tîrîlă Patric-Gabriel – problema 6

select tc.title\_id, tc.status, tc.copy\_id,

CASE

WHEN r.act\_ret\_date is null THEN 'RENTED'

WHEN r.act\_ret\_date is not null THEN 'AVAILABLE'

ELSE 'AVAILABLE'

END

from title\_copy tc, rental r

where r.title\_id = tc.title\_id and r.book\_date = (

select max(rr.book\_date)

from rental rr

where rr.title\_id = tc.title\_id and rr.copy\_id = tc.copy\_id

);

Gavrila Florin-Alexandru – problema 6

select tc.copy\_id, tc.status, nvl2(r.act\_ret\_date, 'AVAILABLE', 'RENTED')

from title\_copy tc, rental r

where r.title\_id = tc.title\_id

and book\_date = (select max(book\_date)

from rental

where copy\_id = tc.copy\_id

and tc.title\_id = title\_id);

Dumitrescul Eduard – problema 6

select t.title\_id, t.title, tc.copy\_id, tc.status, case

when (tc.copy\_id, t.title\_id) in

(select r.copy\_id, r.title\_id

from rental r

join title\_copy tc on tc.title\_id = r.title\_id and tc.copy\_id = r.copy\_id

where r.book\_date = (select max(r2.book\_date)

from rental r2

where r2.title\_id = r.title\_id

and r2.copy\_id = r.copy\_id)

and r.act\_ret\_date is null)

then 'RENTED'

else 'AVAILABLE'

end

from title t, title\_copy tc

where t.title\_id = tc.title\_id;

--PROFA

Select a.title\_id, title, copy\_id, status status\_setat,

case when (a.title\_id, copy\_id) not in (select title\_id, copy\_id

From rental

Where act\_ret\_date is null)

then 'AVAILABLE'

else 'RENTED'

end status\_corect

From title\_copy a, title b

Where a.title\_id = b.title\_id

order by 1,2;

-- Gavrila Florin-Alexandru problema 7a

select count(\*)

from (select tc.title\_id, tc.copy\_id, tc.status, nvl2(r.act\_ret\_date, 'AVAILABLE', 'RENTED')

from title\_copy tc, rental r

where r.title\_id = tc.title\_id

and tc.status != nvl2(r.act\_ret\_date, 'AVAILABLE', 'RENTED')

and book\_date = (select max(book\_date)

from rental

where copy\_id = tc.copy\_id

and tc.title\_id = title\_id));

Dumitrescul Eduard – problema 7a

select count(\*)

from (select t.title\_id, t.title, tc.copy\_id, tc.status status\_setat, case

when (tc.copy\_id, t.title\_id) in

(select r.copy\_id, r.title\_id

from rental r

join title\_copy tc on tc.title\_id = r.title\_id and tc.copy\_id = r.copy\_id

where r.book\_date = (select max(r2.book\_date)

from rental r2

where r2.title\_id = r.title\_id

and r2.copy\_id = r.copy\_id)

and r.act\_ret\_date is null)

then 'RENTED'

else 'AVAILABLE'

end status\_corect

from title t, title\_copy tc

where t.title\_id = tc.title\_id)

where status\_setat != status\_corect;

Tîrîlă Patric-Gabriel – problema 7a

select count(\*)

from (

select tc.title\_id, tc.status status\_1, tc.copy\_id,

CASE

WHEN r.act\_ret\_date is null THEN 'RENTED'

WHEN r.act\_ret\_date is not null THEN 'AVAILABLE'

ELSE 'AVAILABLE'

END status\_2

from title\_copy tc, rental r

where r.title\_id = tc.title\_id and r.book\_date = (

select max(rr.book\_date)

from rental rr

where rr.title\_id = tc.title\_id and rr.copy\_id = tc.copy\_id

)

) tabel

where tabel.status\_1 <> tabel.status\_2;

--7a DOGARU MIHAIL DANUT (partea din with e de la PROFA)

with tabela as(

Select a.title\_id,status status\_setat,

case when (a.title\_id, copy\_id) not in (select title\_id, copy\_id

From rental

Where act\_ret\_date is null)

then 'AVAILABLE'

else 'RENTED'

end status\_corect

From title\_copy a, title b

Where a.title\_id = b.title\_id

order by 1,2)

select count(t.title\_id)

from tabela t

where t.status\_setat != t.status\_corect;

--Gavrila Florin-Alexandru problema 7b

update title\_copy\_gfa

set status =

case when (title\_id, copy\_id) not in (select title\_id, copy\_id

From rental

Where act\_ret\_date is null)

then 'AVAILABLE'

else 'RENTED'

end ;

--Dumitrescul Eduard – problema 7b

update title\_copy\_dev tcd

set status = (

with aux as (select t.title\_id title\_id, t.title, tc.copy\_id copy\_id, tc.status status\_setat, case

when (tc.copy\_id, t.title\_id) in

(select r.copy\_id, r.title\_id

from rental r

join title\_copy tc on tc.title\_id = r.title\_id and tc.copy\_id = r.copy\_id

where r.book\_date = (select max(r2.book\_date)

from rental r2

where r2.title\_id = r.title\_id

and r2.copy\_id = r.copy\_id)

and r.act\_ret\_date is null)

then 'RENTED'

else 'AVAILABLE'

end status\_corect

from title t, title\_copy tc

where t.title\_id = tc.title\_id)

select aux.status\_corect

from aux

where aux.title\_id = tcd.title\_id

and aux.copy\_id = tcd.copy\_id

)

Ilie Florin-Alexandru – problema 9

select m.last\_name, m.first\_name, t.title, count(r.book\_date) as NrImp

from member m

join rental r on m.member\_id = r.member\_id

join title t on r.title\_id = t.title\_id

group by m.last\_name, m.first\_name, t.title;

Ilie Florin-Alexandru-problema 10

select m.last\_name, m.first\_name, t.title, tc.copy\_id, count(r.book\_date)

from member m

join rental r on m.member\_id = r.member\_id

join title t on r.title\_id = t.title\_id

join title\_copy tc on r.title\_id = tc.title\_id

group by m.last\_name, m.first\_name, t.title, tc.copy\_id;

Tîrîlă Patric-Gabriel

CREATE TABLE title\_copy\_tpg AS SELECT \* FROM title\_copy;

update title\_copy\_tpg set status =

CASE

WHEN (title\_id, copy\_id) in (

select title\_id, copy\_id

from rental rr

where rr.act\_ret\_date is null

) THEN 'RENTED'

ELSE 'AVAILABLE'

END;

--Dumitrescul Eduard – problema 8

select rsv.res\_date, rsv.title\_id, case

when rsv.res\_date in

(select rnt.book\_date from rental rnt where rnt.title\_id = rsv.title\_id and rnt.member\_id = rsv.member\_id)

then 'yes'

else 'no'

end

from reservation rsv;

-- Gavrila Florin-Alexandru problema 8

select CASE when re.res\_date in (select r1.book\_date

from rental r1

where r1.member\_id = r.member\_id

and r1.title\_id = r.title\_id) then 'Da' else 'Nu' end luat\_rezervare

from reservation re, member m, rental r

where re.member\_id = m.member\_id

and m.member\_id = r.member\_id

and re.title\_id = r.title\_id;

-- Gavrila Florin-Alexandru problema 9

select m.last\_name, m.first\_name, t.title\_id, t.title, count(\*)

from member m, rental r, title t

where r.member\_id = m.member\_id

and r.title\_id = t.title\_id

group by m.last\_name, m.first\_name, t.title\_id, t.title

order by 1;

-- Dumitrescul Eduard – problema 9

select m.last\_name, m.first\_name, t.title\_id,

(select count(\*)

from rental r

where r.member\_id = m.member\_id and r.title\_id = t.title\_id) NR

from member m, title t

order by m.last\_name, m.first\_name;

-- Dumitrescul Eduard – problema 10

select m.last\_name, m.first\_name, tc.title\_id, tc.copy\_id,

(select count(\*)

from rental r

where r.member\_id = m.member\_id and r.title\_id = tc.title\_id and r.copy\_id = tc.copy\_id) NR

from member m, title\_copy tc

order by m.last\_name, m.first\_name;

-- Gavrila Florin-Alexandru problema 10

select m.last\_name, m.first\_name, t.title\_id, tc.copy\_id, t.title, count(\*)

from member m, rental r, title t, title\_copy tc

where r.member\_id = m.member\_id

and r.title\_id = t.title\_id

and tc.title\_id = t.title\_id

and tc.copy\_id = r.copy\_id

group by m.last\_name, m.first\_name, t.title\_id, tc.copy\_id, t.title

order by 1;

--Gavrila Florin-Alexandru problema 11

SELECT r.title\_id, r.copy\_id, tc.status, COUNT(\*) AS rental\_count

FROM title\_copy tc, rental r

where r.title\_id = tc.title\_id

and r.copy\_id = tc.copy\_id

GROUP BY r.title\_id, r.copy\_id, tc.status

HAVING COUNT(\*) = (

SELECT MAX(rental\_count)

FROM (

SELECT r1.title\_id, r1.copy\_id, COUNT(\*) AS rental\_count

FROM title\_copy tc1

JOIN rental r1 ON tc1.title\_id = r1.title\_id AND tc1.copy\_id = r1.copy\_id

GROUP BY r1.title\_id, r1.copy\_id

) subquery

WHERE subquery.title\_id = r.title\_id

);

-- Dumitrescul Eduard-Valentin – problema 11

select tc.title\_id, tc.copy\_id, (select count(\*) from rental r where r.title\_id = tc.title\_id and r.copy\_id = tc.copy\_id) no\_rented, tc.status

from title\_copy tc

where (tc.title\_id, ((select count(\*) from rental r where r.title\_id = tc.title\_id and r.copy\_id = tc.copy\_id)))

in (select tc.title\_id ,

max((select count(\*) from rental r where r.title\_id = tc.title\_id and r.copy\_id = tc.copy\_id))

from title\_copy tc

group by tc.title\_id);

-- Gavrila Florin-Alexandru problema 12

select count(\*)

from rental

where to\_char(book\_date, 'dd') = '01'

or to\_char(book\_date, 'dd') = '02';

select count(\*)

from rental

where to\_char(book\_date, 'mm') = to\_char(sysdate, 'mm');

-- Dumitrescul Eduard – problema 12 (a, b, c)

with days as (select to\_date(trunc(sysdate, 'mm'),'dd.mm.yy')+level-1 day\_date

from dual

connect by level <= 2)

select d.day\_date,

(select count(\*) from rental r where to\_char(r.book\_date, 'dd-mm-yyyy') = to\_char(d.day\_date, 'dd-mm-yyyy')) cnt

from days d;

select r.book\_date, count(\*) no\_rented

from rental r

where r.book\_date >= to\_date(to\_char(sysdate, 'mm-yyyy'), 'mm-yyyy')

group by r.book\_date;

with days as (select to\_date(trunc(sysdate, 'mm'),'dd.mm.yy')+level-1 day\_date

from dual

connect by level <= TO\_CHAR(LAST\_DAY(sysdate),'DD'))

select to\_char(d.day\_date, 'dd-mm-yyyy'),

(select count(\*) from rental r where to\_char(r.book\_date, 'dd-mm-yyyy') like to\_char(d.day\_date, 'dd-mm-yyyy')) cnt

from days d;

**Laborator 3**

--9.

select m.member\_id, first\_name, last\_name,

t.title\_id, t.title, count(r.title\_id)

from member m, title t, rental r

where t.title\_id = r.title\_id (+)

and m.member\_id(+) = r.member\_id

group by m.member\_id, first\_name, last\_name, t.title\_id, title

order by 1,4;

--11

with info as

(select t.title\_id, max(title), c.copy\_id, max(status), count(\*) nr

from title t, title\_copy c, rental r

where t.title\_id = c.title\_id

and c.copy\_id = r.copy\_id

and c.title\_id = r.title\_id

group by t.title\_id, c.copy\_id)

select \*

from info i

where nr = (select max(nr)

from info

where title\_id = i. title\_id)

order by 1;

-- 12 – dumitrescul eduard

with days as (select to\_date(trunc(sysdate, 'mm'),'dd.mm.yy')+level-1 day\_date

from dual

connect by level <= TO\_CHAR(LAST\_DAY(sysdate),'DD'))

select to\_char(d.day\_date, 'dd-mm-yyyy'),

(select count(\*) from rental r where to\_char(r.book\_date, 'dd-mm-yyyy') like to\_char(d.day\_date, 'dd-mm-yyyy')) cnt

from days d;

--4.

DECLARE

v\_dep departments.department\_name%TYPE;

BEGIN

SELECT department\_name

INTO v\_dep

FROM employees e, departments d

WHERE e.department\_id=d.department\_id

GROUP BY department\_name

HAVING COUNT(\*) <> (SELECT MAX(COUNT(\*))

FROM emp0\_prof

GROUP BY department\_id);

DBMS\_OUTPUT.PUT\_LINE('Departamentul '|| v\_dep);

exception

when too\_many\_rows then

DBMS\_OUTPUT.PUT\_LINE('mai multe linii!');

when no\_data\_found then

DBMS\_OUTPUT.PUT\_LINE('nicio linie!');

END;

/

Exercitiul 6 Liviu

VARIABLE rezultat VARCHAR2(35)

declare

numar number;

BEGIN

SELECT department\_name,count(\*)

INTO

:rezultat,numar

FROM

employees e, departments d

WHERE e.department\_id=d.department\_id

GROUP BY department\_name

HAVING COUNT(\*) = (SELECT MAX(COUNT(\*))

FROM

employees

GROUP BY department\_id);

DBMS\_OUTPUT.PUT\_LINE('Departamentul '|| :rezultat||' numarul '|| numar );

END;

/

PRINT rezultat

Ex. 6 – Papusoi Rares

VARIABLE rezultat VARCHAR2(35)

VARIABLE rezultat2 VARCHAR2 (35)

BEGIN

SELECT department\_name, count(\*)

INTO :rezultat, :rezultat2

FROM employees e, departments d

WHERE e.department\_id=d.department\_id

GROUP BY department\_name

HAVING COUNT(\*) = (SELECT MAX(COUNT(\*))

FROM employees

GROUP BY department\_id);

DBMS\_OUTPUT.PUT\_LINE('Departamentul '|| :rezultat);

DBMS\_OUTPUT.PUT\_LINE('Angajati: ' || :rezultat2);

END;

/

-- ex 2 Dumitrescul Eduard

with days as

(select to\_date('01.10.2023', 'DD.MM.YYYY') + level - 1 day\_date from dual

connect by level <= to\_char(LAST\_DAY(to\_date('01.10.2023', 'DD.MM.YYYY')), 'DD'))

select to\_char(d.day\_date, 'dd-mm-yyyy'),

(select count(\*) from rental r where to\_char(r.book\_date, 'dd-mm-yyyy') like to\_char(d.day\_date, 'dd-mm-yyyy')) cnt

from days d;

--2 b

Create table octombrie\_prof

( id number(5), data date);

Declare

contor number(3) := 0;

Data\_aux date := ‘01-OCT-2023';

Begin

Loop

insert into octombrie\_prof

values (contor, data\_aux+contor);

contor := contor+1;

exit when contor = 31;

End loop;

End;

/

**Laborator 4**

select employee\_id

bulk collect into t

from

(select employee\_id

from employees

where commission\_pct is null

order by salary)

where rownum<=5;

Tîrîlă Patric-Gabriel ex 1

DECLARE

type colectie is table of employees.employee\_id%TYPE index by pls\_integer;

v\_colectie colectie;

BEGIN

select employee\_id

bulk collect into v\_colectie

from (

select employee\_id

from employees

where commission\_pct is null

order by salary asc

)

where rownum <= 5;

for i in v\_colectie.first .. v\_colectie.last loop

update employees

set salary = salary + (5/100 \* salary)

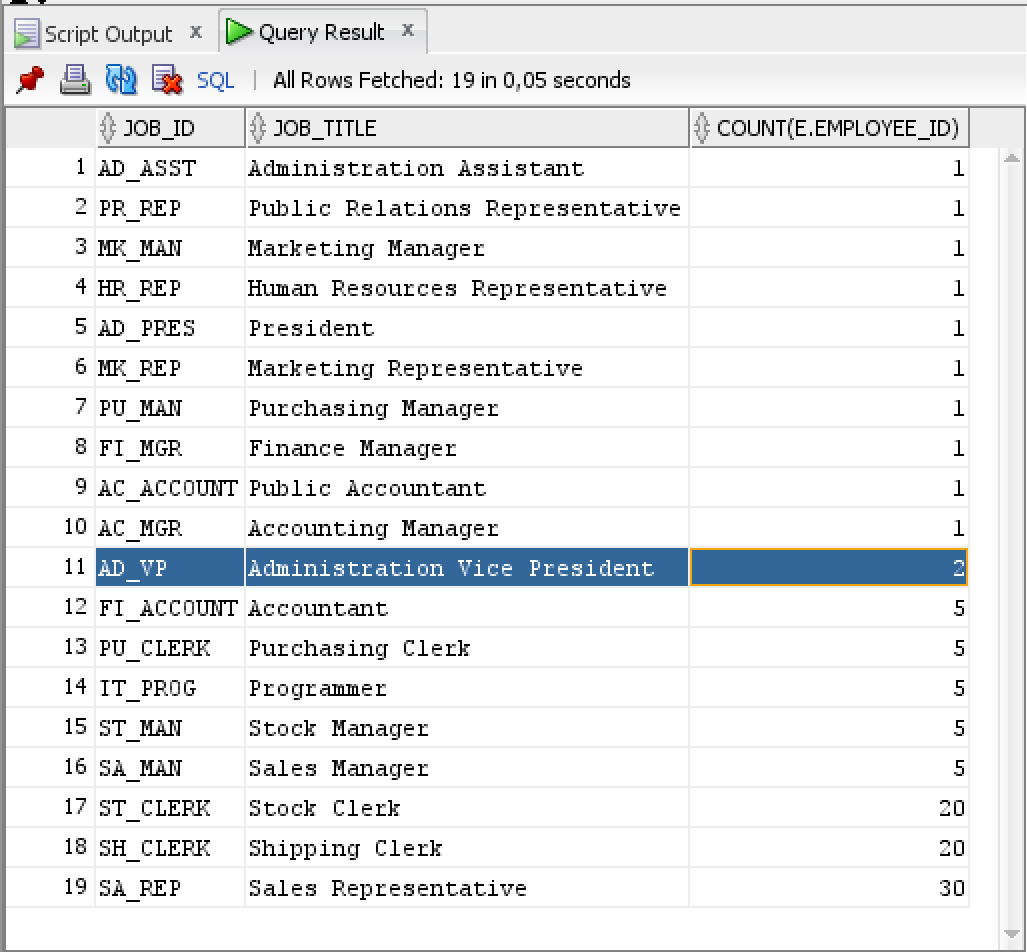
where employee\_id = v\_colectie(i);

end loop;

END;

**Laborator 5 – PLSQL 3**

--e1



--a. Cursor explicit

DECLARE  
 CURSOR cur\_jobs IS  
 SELECT DISTINCT job\_id AS job\_title FROM employees;  
  
 v\_job employees.job\_id%TYPE;  
 v\_employee\_name employees.first\_name%TYPE;  
 v\_employee\_salary employees.salary%TYPE;  
 BEGIN  
 FOR job\_rec IN cur\_jobs LOOP  
 v\_job := job\_rec.job\_title;  
 DBMS\_OUTPUT.PUT\_LINE('Job: ' || v\_job);  
  
 IF v\_job IS NOT NULL THEN  
 FOR emp\_rec IN (SELECT first\_name, salary FROM employees WHERE job\_id = v\_job) LOOP  
 v\_employee\_name := emp\_rec.first\_name;  
 v\_employee\_salary := emp\_rec.salary;  
 DBMS\_OUTPUT.PUT\_LINE(' Employee: ' || v\_employee\_name || ', Salary: ' || v\_employee\_salary);  
 END LOOP;  
 ELSE  
 DBMS\_OUTPUT.PUT\_LINE(' No employees found for this job.');  
 END IF;  
 END LOOP;  
 END;  
 /

--profa

DECLARE

CURSOR c1 IS

SELECT job\_title, job\_id

FROM jobs;

CURSOR c2 IS

SELECT last\_name, salary, job\_id

FROM employees;

v\_titlu jobs.job\_title %TYPE;

v1\_id jobs.job\_id %TYPE;

v\_nume employees.last\_name %TYPE;

v\_salary employees.salary %TYPE;

v2\_id employees.job\_id %TYPE;

BEGIN

OPEN c1;

LOOP

FETCH c1 into v\_titlu, v1\_id;

EXIT WHEN c1%NOTFOUND;

DBMS\_OUTPUT.put\_line('Id-ul jobului : '

|| v1\_id

|| ' , '

|| v\_titlu);

OPEN c2;

LOOP

FETCH c2 into v\_nume, v\_salary, v2\_id;

EXIT WHEN c2%NOTFOUND;

IF v1\_id = v2\_id THEN

DBMS\_OUTPUT.put\_line('Angajatul cu numele '

|| v\_nume || ' avand salariul: '

|| v\_salary);

END IF;

END LOOP;

close c2;

END LOOP;

close c1;

END;

/

--b. Ciclu cursor

--c. Ciclu cursor cu subcereri

-- Tirila Patric-Gabriel

declare

v\_rows\_found BOOLEAN := FALSE;

begin

for i in (

select job\_title titlu, job\_id id

from jobs

) loop

v\_rows\_found := FALSE;

DBMS\_OUTPUT.PUT\_LINE('Jobul ' || i.titlu);

for j in (

select last\_name lname, salary salary

from employees

where job\_id = i.id

) loop

v\_rows\_found := TRUE;

DBMS\_OUTPUT.PUT\_LINE('Employee: ' || j.lname || ' ' || j.salary);

end loop;

if v\_rows\_found = FALSE

then DBMS\_OUTPUT.PUT\_LINE('Nu are angajati');

end if;

end loop;

end;

/

--d. Expresii cursor